

Efficient Bayesian Community Detection using Non-negative Matrix Factorisation

Ioannis Psorakis ^{*†}, Stephen Roberts [‡] and Ben Sheldon [§]

September 28, 2010

Abstract

Identifying overlapping communities in networks is a challenging task. In this work we present a novel approach to community detection that utilises the Bayesian non-negative matrix factorisation (NMF) model to produce a probabilistic output for node memberships. The scheme has the advantage of computational efficiency, soft community membership and an intuitive foundation. We present the performance of the method against a variety of benchmark problems and compare and contrast it to several other algorithms for community detection. Our approach performs favourably compared to other methods at a fraction of the computational costs.

Keywords: Community detection, non-negative matrix factorisation, Bayesian inference.

1 Introduction

The network paradigm is widely used to model real-world complex systems, by focusing on the pattern of associations between their structural components. A system is captured as a mathematical graph, where nodes (or vertices) denote the presence of an entity and edges (or links) signify some sort of association (or interaction). In contrast to other data manipulation approaches where each element is described by a set of attributes (for example $\mathbf{x} \in \mathbb{R}^D$), here our data is captured in a *relational* form and inferences are made primarily based on their connectivity patterns.

Real-world networks differ from the classic Erdos-Renyi random graph because the presence of a link between two nodes is not generated by a Bernoulli trial with same success probability across all possible pairs. Instead, real-world networks exhibit an inhomogeneous distribution of edges among vertices [12], creating ‘hotspots’ of heightened connectivity. These *modules* or *communities* are densely connected, relatively independent compartments [12] [20] of the network that account for its form and function as a system [26]. The intuition behind that *mesoscopic* organisation of networks is intuitively straightforward, with many examples from everyday life; human social networks consist of cliques of friends, Web pages can be grouped into collections with similar topic, etc.

As an example, consider the simple undirected graph of Fig. 1, described by an adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ where we can immediately distinguish the two densely connected compartments C1 and C2. As we

^{*}Authors in alphabetic order.

[†]Pattern Analysis & Machine Learning Research Group, Department of Engineering Science, University of Oxford. Parks Road, Oxford, OX1 3PJ U.K. Email yannis@robots.ox.ac.uk

[‡]Pattern Analysis & Machine Learning Research Group. Email sjrob@robots.ox.ac.uk

[§]Dept Zoology, University of Oxford

are not sure of the membership of node 5, it is fairly reasonable to consider it as an *overlap* of C1 and C2. Therefore, we can express our community partition as an expansion of our network to a bipartite graph with incidence matrix $\mathbf{B} \in \mathbb{R}^{N \times K}$ so that $b_{ik} = 1$ denotes that node i belongs to a group k and is zero otherwise. Although the human eye is an excellent analytic tool for simple visualised data [24], the algorithmic process of identifying the number of groups, classifying their members and spotting overlaps in any given network is far from straightforward.

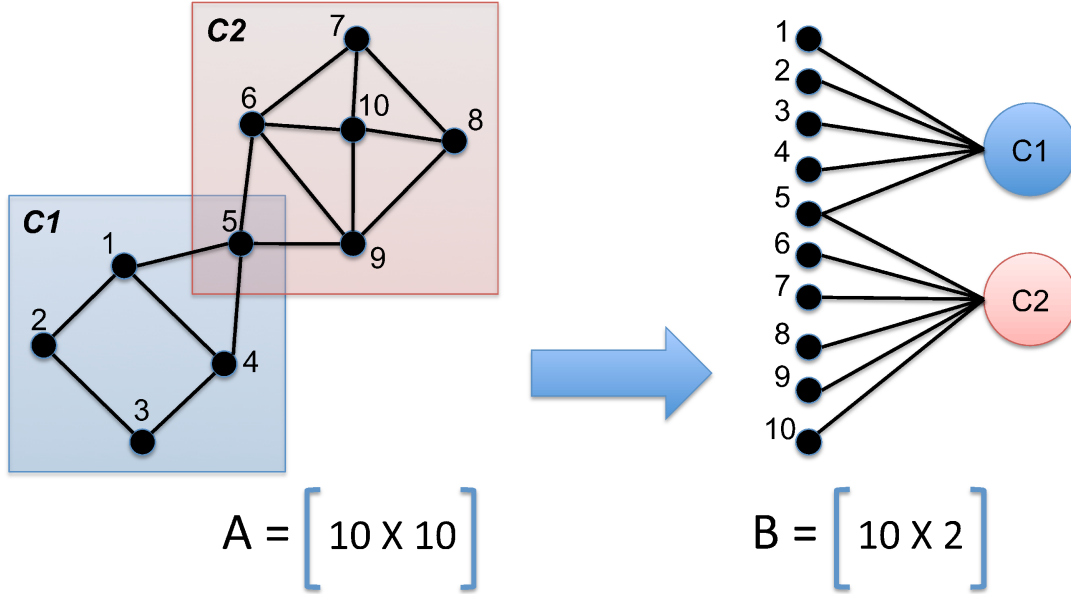


Figure 1: The problem of community detection can be viewed as mapping our network to a bipartite graph, where the first class corresponds to individual nodes and the second to communities. Links connecting nodes to multiple communities help us capture *overlapping* phenomena, in which an individual can participate in many groups.

Extracting community structure from a network is a considerably challenging task [26], both as an inference and as a computational efficiency problem. One of the main reasons is that there is no formal, application-independent definition of what a community actually is [12]; we simply accept that communities are node subsets with a ‘statistically surprising’ link density, usually measured ‘modularity’ Q from Mark Newman and Michelle Girvan [27]. Nevertheless, this definition lies in the heart of modern community detection algorithms, manifested either as an exploration of configurations (such as \mathbf{B} described above) that seek to approximately maximise Q (as direct optimisation is NP-hard [12]), or other techniques that exploit characteristics of the network topology to group together nodes with high mutual connectivity (and use Q as a performance measure). A fairly comprehensive review of existing methods is provided in [30] and [12], while [9] provides a table summarising the computational complexity of popular algorithms.

The most significant drawbacks of modern community detection algorithms are *hard partitioning* or/and *computational complexity*. Many widely popular methods such as Girvan-Newman [27], Extremal Optimisation [10] or Spectral Partitioning [25] cannot account for the *overlapping* nature of communities, which is an important characteristic of complex systems [12]. On the other hand, the Clique Percolation Method (CPM) [28] allows node assignment to multiple modules but does not provide some ‘degree of belief’ on how strongly an individual belongs to a certain group. Therefore, our aim is not only to model group overlaps but also capture the *likelihood* of node memberships in a disciplined way, expressing each row of the incidence matrix \mathbf{B}

as a *membership distribution* over communities. Additionally, we want to avoid the computational complexity issues of traditional combinatorial methods.

Towards these goals, community detection can be seen as a *generative model* in a probabilistic framework. This has the advantage that, in principle, fully Bayesian models may be formulated in which *priors* exist over all the model parameters. This enables, for example, model selection (to determine e.g. how many communities there are) and the principled handling of uncertainty, noise and missing data.

In this paper we consider the case in which constraints exist in our beliefs regarding the generative process for the observed data. In particular we investigate the issue of enforcing *positivity* onto the model. As the model we consider is evaluated in a Bayesian manner, model selection may be applied to infer the complexity of representation in the solution space of inferred communities.

This paper is organised as follows. We first introduce the basic concepts of the theory and our data decomposition goals. Details of the Bayesian paradigm under which model inference is performed are then presented. Representative results are given in the next sections followed by conclusions and discussion.

2 Methods

We consider a matrix of observed interactions between a set of N *atoms*, which in the context of community detection we consider to be individuals. We consider an *interaction matrix* denoted $\mathbf{V} \in \mathbb{R}_+^{N \times N}$ such that v_{ij} represents a count process detailing the number of interactions between atoms (individuals) i and j .¹

2.1 Decomposition

We consider the decomposition of the observed interaction matrix \mathbf{V} as a linear combination of K canonical communities, each of which can be seen as a latent (hidden) generator of interactions between atoms. Hence,

$$\hat{v}_{ij} = \sum_{k=1}^K w_{ik} h_{kj}, \quad (1)$$

in which we regard the w_{ik} as mixing coefficients and the h_{kj} as elements forming a basis set of community structures. The above equation may be re-written in matrix form by defining $\mathbf{W} \in \mathbb{R}^{N \times K}$ and $\mathbf{H} \in \mathbb{R}^{K \times N}$

$$\hat{\mathbf{V}} = \mathbf{WH}. \quad (2)$$

Without constraint, Equation 2 is ill-posed, i.e. an infinite number of equivalent solutions exist. Many well-known decomposition methods can be seen as members of a family of approaches which impose constraints in the solution space to make the matrix-product decomposition well-posed.

PCA: Principal Component Analysis [17] avoids the problem of an ill-posed solution space by making several constraints. The first is that the observed data distribution and the basis are Gaussian distributed (in that only second-order statistics are employed in the PCA formalism) and that the basis is orthogonal. This still leaves a permutation degree of freedom, which is removed by sorting the basis in order of variance (in effect an ordering of the eigenvalues).

¹A brief note on nomenclature - as the method described here has its roots in the work in *non-negative matrix factorization* (NMF), we keep to the historic nomenclature of the latter to make easier access for the reader who wishes to follow up primary references.

ICA: Independent Component Analysis [8, 31] makes the solutions to Equation 2 well-posed by forcing statistical independence between the components of the basis without making strong assumptions regarding the Gaussianity of the component distributions, unlike PCA. This gives rise to a methodology which allows projective decompositions similar to PCA for non-Gaussian data.

NMF: Non-negative Matrix Factorization makes the assumption that all the elements of matrices \mathbf{V} , \mathbf{W} , \mathbf{H} lie in \mathbb{R}_+ . This is enough, up to an arbitrary scaling degree of freedom, to ensure that solutions to Equation 2 are well-posed. The latter assumptions of non-negativity match our prior beliefs regarding the generation of the observed interaction count data, \mathbf{V} , and we extend our discussion of NMF in the following sections.

2.2 Generative Model

We consider the observed data to be modelled as a Poisson process with expectations given by the elements of $\hat{\mathbf{V}} = \mathbf{W}\mathbf{H}$. This Poisson model lies at the core of the NMF methodology first developed by Lee & Seung [21]. The standard maximum-likelihood solution is to find \mathbf{W} , \mathbf{H} such that $p(\mathbf{V}|\mathbf{W}, \mathbf{H})$ is maximized, or alternately, the energy function $-\log p(\mathbf{V}|\mathbf{W}, \mathbf{H})$ is minimized.

Consider an element v of \mathbf{V} and the associated expected counts from the model, \hat{v} . The negative log likelihood of v under a Poisson model is:

$$-\log p(v|\hat{v}) = -v \log \hat{v} + \hat{v} + \log v!. \quad (3)$$

Using the Stirling approximation to second order, namely

$$\log v! = v \log v - v + \frac{1}{2} \log(2\pi v), \quad (4)$$

so Equation 3 can be written as,

$$-\log p(v|\hat{v}) = v \log \left(\frac{v}{\hat{v}} \right) + \hat{v} - v + \frac{1}{2} \log(2\pi v) \quad (5)$$

and the full negative log-likelihood for all the observed data as

$$-\log p(\mathbf{V}|\hat{\mathbf{V}}) = -\sum_i \sum_j \log p(v_{ij}|\hat{v}_{ij}). \quad (6)$$

2.2.1 Shrinkage hyperparameters

As each of the $k \in \{1 \dots K\}$ columns of \mathbf{W} and rows of \mathbf{H} represents the contribution from a single latent community (as per Equation 1) we allow for different shrinkage hyperparameters, defined as the set of $\{\beta_k\}$. Following the development of this model in [33] and similar models for probabilistic PCA [34] and ICA [7, 32] we place independent half-normal priors over the columns of \mathbf{W} and rows of \mathbf{H} in which the β_k may be seen as precision (inverse variance) parameters:

$$\begin{aligned} p(w_{ik}|\beta_k) &= \mathcal{HN}(w_{ik}|0, \beta_k^{-1}) \\ p(h_{kj}|\beta_k) &= \mathcal{HN}(h_{kj}|0, \beta_k^{-1}), \end{aligned} \quad (7)$$

where

$$\mathcal{HN}(x|0, \beta^{-1}) = \sqrt{\frac{2}{\pi}} \beta^{1/2} \exp\left(-\frac{1}{2}\beta x^2\right). \quad (8)$$

Defining the vector β as $[\beta_1, \dots, \beta_K]$, this leads to negative log priors over \mathbf{W} and \mathbf{H} as:

$$\begin{aligned} -\log p(\mathbf{W}|\beta) &= \sum_i \sum_k \frac{1}{2} \beta_k w_{ik}^2 - \frac{F}{2} \log \beta_k + \text{const}, \\ -\log p(\mathbf{H}|\beta) &= \sum_k \sum_j \frac{1}{2} \beta_k h_{kj}^2 - \frac{N}{2} \log \beta_k + \text{const}. \end{aligned} \quad (9)$$

The net effect of β on the elements of \mathbf{W} and \mathbf{H} may be considered as follows. As the negative log probability may be regarded as an error or energy function our goal is to descend its surface to a point of minimum. Consider the negative derivative w.r.t. a single element of \mathbf{W} or \mathbf{H} of the above Equations, e.g.

$$-\frac{\partial (-\log p(\mathbf{W}|\beta))}{\partial w_{ik}} = -\beta_k w_{ik}. \quad (10)$$

Incremental changes in w_{ik} , as we iterate towards a solution, are proportional to the negative gradient of the energy function, so the effect of the prior is to promote a *shrinkage* to zero of w_{ik} with a rate constant proportional to β_k . A large β_k represents a belief that the half-normal distribution over w_{ik} has small variance, and hence w_{ik} is expected to lie close to zero. As we shall see, the priors and the likelihood function (quantifying how well we explain the data) are combined with the net effect that columns of \mathbf{W} (and rows of \mathbf{H}) which have little effect in changing how well we explain the observed data will shrink close to zero. This generic approach is well known in the statistics literature, as *shrinkage* or *ridge regression* [2] and in the machine learning community as *automatic relevance determination* [4].

Finally we must place prior distributions over the β_k . We assume the set of β_k are independent² and as these are scale hyperparameters we place a standard Gamma distribution over them [2]:

$$p(\beta_k|a_k, b_k) = \frac{b_k^{a_k}}{\Gamma(a_k)} \beta_k^{a_k-1} \exp(-\beta_k b_k), \quad (11)$$

in which the hyper-hyperparameters a_k, b_k defining the gamma distribution over β_k are fixed. The negative log of the probability distribution over β is hence,

$$-\log p(\beta) = \sum_k [\beta_k b_k - (a_k - 1) \log \beta_k] + \text{const}. \quad (12)$$

2.2.2 Overall posterior cost function

Figure 2 shows the generative graphical model for the NMF method. The joint distribution over all variables is

$$p(\mathbf{V}, \mathbf{W}, \mathbf{H}, \beta) = p(\mathbf{V}|\mathbf{W}, \mathbf{H}) p(\mathbf{W}|\beta) p(\mathbf{H}|\beta) p(\beta), \quad (13)$$

and the model posterior over all parameters, given the observations is:

$$p(\mathbf{W}, \mathbf{H}, \beta|\mathbf{V}) = \frac{p(\mathbf{V}, \mathbf{W}, \mathbf{H}, \beta)}{p(\mathbf{V})}. \quad (14)$$

²This corresponds to the belief that the existence of one community is not dependent upon others. Clearly, there will be situations in which this can be extended to allow for a full inter-dependency between communities. We do not consider this here, however. Allowing dependency is similar to the notion of *structure priors* discussed in [29].

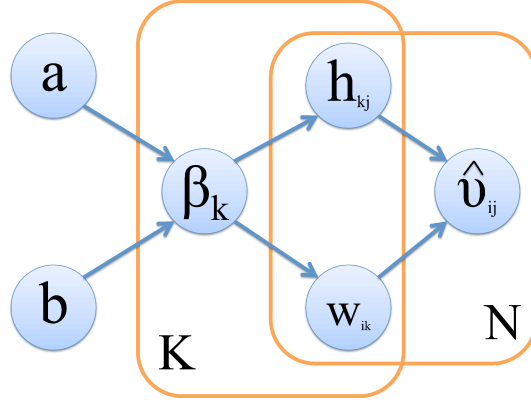


Figure 2: Graphical model showing the generation of count processes, \mathbf{V} from the latent structure \mathbf{W} and \mathbf{H} the components of which have scale hyperparameters β . The hyper-hyperparameters a, b are fixed in the model.

Noting that $p(\mathbf{V})$ is a constant w.r.t. the inference over the model's free parameters, we aim hence to maximize the model posterior given the observations. This is equivalent to minimizing the negative log posterior, which we may regard as an energy (or error) function, \mathcal{U} say. We hence define

$$\mathcal{U} = \log p(\mathbf{V}|\mathbf{W}, \mathbf{H}) - \log p(\mathbf{W}|\beta) - \log p(\mathbf{H}|\beta) - \log p(\beta). \quad (15)$$

Expanding this expression using the results from Equations 5, 6, 9 and 12 and collating all terms independent of the model parameters into a constant, gives:

$$\begin{aligned} \mathcal{U} &= \sum_i \sum_j \left[v_{ij} \log \left(\frac{v_{ij}}{\hat{v}_{ij}} \right) + \hat{v}_{ij} \right] \\ &+ \frac{1}{2} \sum_k \left[\left(\sum_f \beta_k w_{fk}^2 \right) + \left(\sum_n \beta_k h_{kn}^2 \right) - (F + N) \log \beta_k \right] \\ &+ \sum_k [\beta_k b_k - (a_k - 1) \log \beta_k] + \text{const.} \end{aligned} \quad (16)$$

2.3 Parameter Inference

There are a variety of approaches one could take to infer $\mathbf{W}, \mathbf{H}, \beta$ given Equation 16. In this paper we follow [21, 22, 3, 33] and utilize a rapid fixed point maximum a posteriori (MAP) algorithm which guarantees to preserve the non-negativity of all parameters in the model. At each iteration the following re-estimations are made,

$$\mathbf{H} \leftarrow \left(\frac{\mathbf{H}}{\mathbf{W}^\top \mathbf{1} + \mathbf{B}\mathbf{H}} \right) \mathbf{W}^\top \left(\frac{\mathbf{V}}{\mathbf{W}\mathbf{H}} \right) \quad (17)$$

$$\mathbf{W} \leftarrow \left(\frac{\mathbf{W}}{\mathbf{1}\mathbf{H}^\top + \mathbf{W}\mathbf{B}} \right) \left(\frac{\mathbf{V}}{\mathbf{W}\mathbf{H}} \right) \mathbf{H}^\top \quad (18)$$

in which we define \mathbf{B} as having the elements β_k along its diagonal and zeros elsewhere and $\mathbf{1}$ is a vector of ones. We keep to the notation convention that $\left(\frac{\mathbf{X}}{\mathbf{Y}} \right)$ represents *element-by-element* division and does *not*

represent \mathbf{XY}^{-1} . The values of β_k are re-estimated by setting to zero the derivative w.r.t. the β_k of the energy function in Equation 16. This gives an estimate for β_k as

$$\beta_k \leftarrow \frac{N + a_k - 1}{\frac{1}{2} \left(\sum_f w_{ik}^2 + \sum_n h_{kj}^2 \right) + b_k} \quad (19)$$

which is the same as the update equation detailed in [33]. The algorithm proceeds by cycling through Equations 17, 18, 19 until a convergence criterion or maximum number of iterations is reached. We note that a fully Bayesian approach to NMF is developed using variational inference in [6]. This offers certain potential improvements over the maximum a posteriori (MAP) solution at the expense of computational speed. The latter we regard as a very important feature of any algorithm for community detection and the current emphasis of our work is directed by computational efficiency.

2.4 Probabilistic community membership

As we may write the observed data as a linear combination of community basis structures and the mixing fractions are strictly non-negative, the model is identical to a *mixture model* in which the elements w_{ik} denote the relative importance of community k in explaining the observed interactions associated with member i . Under the assumption that the i -th member's interactions are explained by some community memberships, it is reasonable to define degrees of community membership, π_{ik} , which sum to unity for each member, as:

$$\pi_{ik} = \frac{w_{ik}}{\sum_{k'} w_{ik'}}. \quad (20)$$

A greedy community allocation scheme for member i is easily achieved, if desired, by choosing the community k^* which is the argmax of either the w_{ik} or π_{ik} .

3 Results

In this section we demonstrate the performance of NMF-based community detection against a variety of benchmark problems. We start with a toy network to illustrate the intuition behind our clustering methodology using graphical examples. Afterwards, we continue by testing our method against artificial problems with observed community structure. Finally, we test our method against popular real-world networks of various sizes and levels of community cohesiveness.

3.1 Initialization

In all the results presented in this paper we allow the maximum number of possible communities K to equal N , the number of members. The hyper-hyperparameters, a and b , which govern the scale of the shrinkage hyperparameters, β_k , are fixed at $a = 1, b = 2$ so that β_k all have vague distributions over them. The initial matrices \mathbf{W}, \mathbf{H} each have elements drawn independently at random from a uniform distribution in the interval $[0, 1]$.

The interaction matrix \mathbf{V} we use for NMF is derived from the (weighted) adjacency matrix of each network with diagonal elements the strengths of each node (the sum of each row or column of the adjacency matrix).

3.2 An Illustrative Example

Consider the simple toy graph of Fig. 3 with $N = 16$ nodes and $M = 25$ edges of varying weights. We extract the mesoscopic (community) structure of this network using NMF, along with the popular Extremal Optimisation (EO) [10], Spectral Partitioning (SP) [25] and Weighted Clique Percolation Method (wCPM) [11].

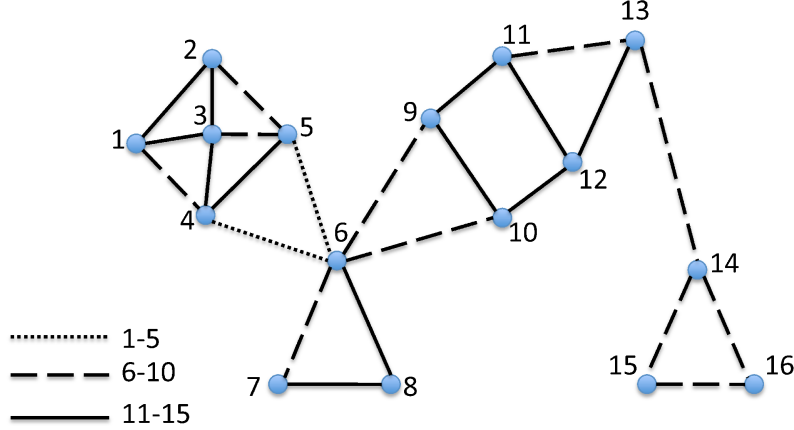


Figure 3: An undirected weighted toy graph with 16 nodes. Each pair of nodes has a different interaction strength as denoted by the different lines.

Although a trivial problem at first glance, each community detection method we applied yielded different modules and node allocations, as seen in Fig. 4. Hard-partitioning methods such as EO and SP produce such inconsistencies mainly due to the ‘broker’ nature of nodes such as 6, 9 or 10, which lie on high-flow paths in the network, making them difficult to assign on one module or the other [12]. Although this issue is addressed by wCPM, which allows node membership to multiple modules, it does not provide some measure of ‘participation strength’ or ‘degree of belief in membership’.

In NMF, communities are viewed as *basis structures*, captured in the model as the K columns of our basis matrix \mathbf{W} (see Section 2). In this framework, we consider each basis structure or community k to have a total binding energy that is allocated to the atoms (nodes) based on $\mathbf{w}_k \in \mathbb{R}_+^{N \times 1}$. For example in Figure 5, we take a column of \mathbf{W} and draw a colormap (left frame) based on the intensity of its elements. Components with non-zero energy correspond to nodes that participate in such basis structure and form a subset of the whole network (right frame). We can see that this basis community in Fig. 5 is dominated by nodes 6, 7 and 8, which contribute most of the binding energy, while the peripheral nodes 4, 5, 9, 10 have some minor participation.

We applied NMF to our synthetic graph, where we extracted $K^* = 4$ communities (bases to which at least one member is allocated) as seen from the four numbered plates in Figure 6. For illustrative purposes, we assigned nodes to communities using greedy allocation, i.e. we put the individual to the community into which it contributes most of its energy. The contribution of each atom i to the total binding energy of each structure k , as denoted by $\mathbf{w}_i \in \mathbb{R}_+^{1 \times K}$, can be viewed as the individual’s *degree of participation* or, when normalised, *probability of membership* to that community. From our example, in Figure 7 we show the different membership distributions for four different nodes in the graph. In accordance to our intuition, we see that node 6, which acts as a mediator between different communities has a more entropic membership distribution while nodes such as 4 or 14 have more confident assignments.

We also can view real-world systems, such as social networks, under the framework we described above; groups of individuals are structures bound together with a given energy (time spent together, genetic relatedness,

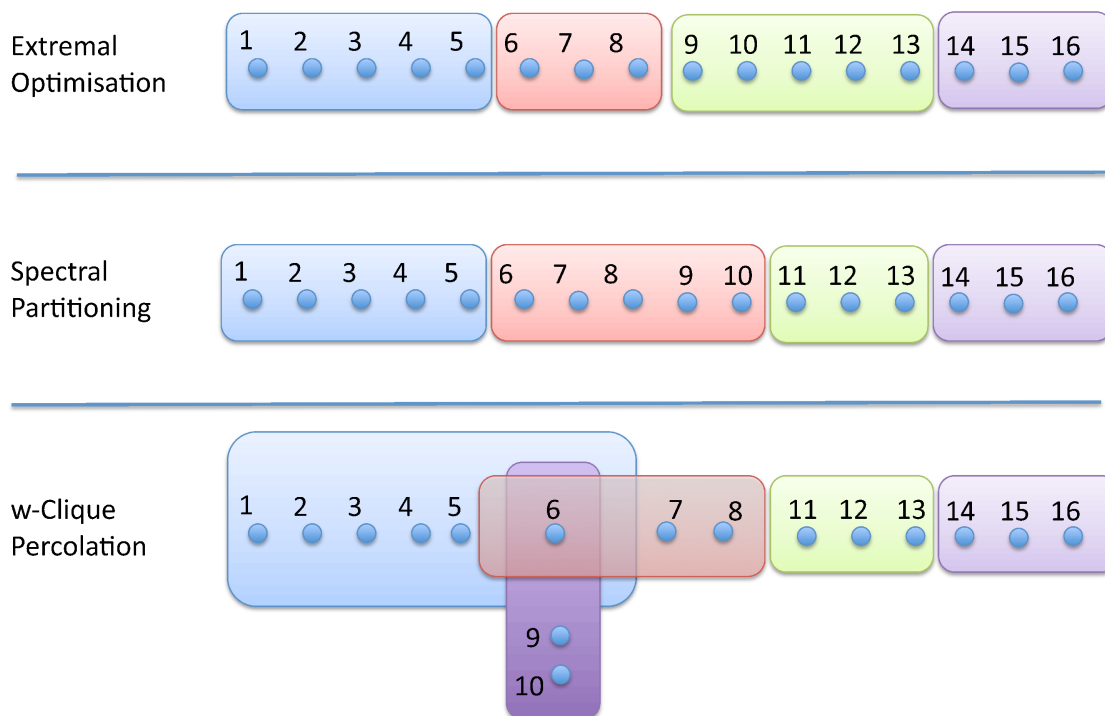


Figure 4: Node allocations to communities for three different community detection methodologies.

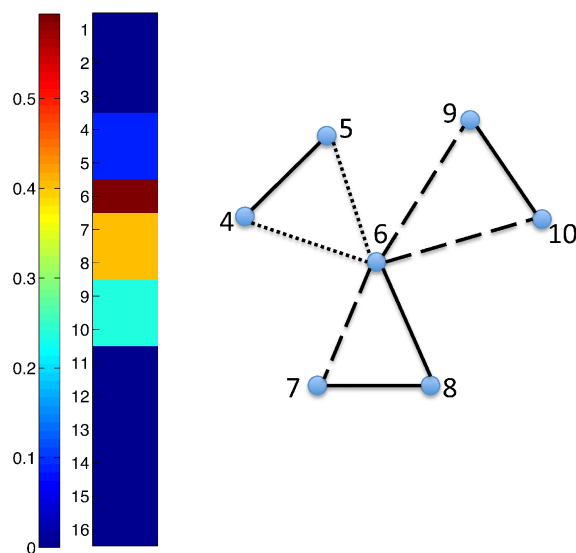


Figure 5: One of the NMF *basis structures*, as extracted from our toy graph. Each atom has a different degree of participation, as it can be seen from the colourmap on the left. Node 6 is a focal individual, contributing the most energy to the structure along with nodes 7 and 8, while nodes 4,5,9 and 10 are peripheral.

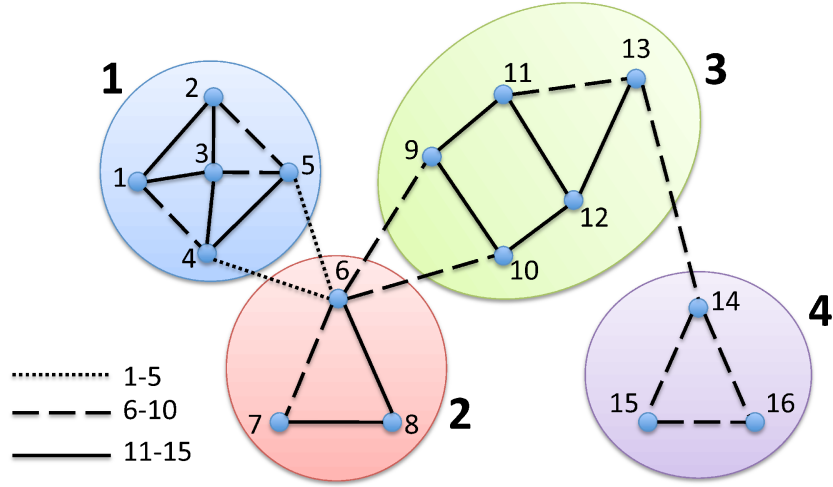


Figure 6: The community structure of our toy graph, where each coloured plate represents a different basis structure. For purposes of illustration, we assigned each node to the community with the highest membership probability.

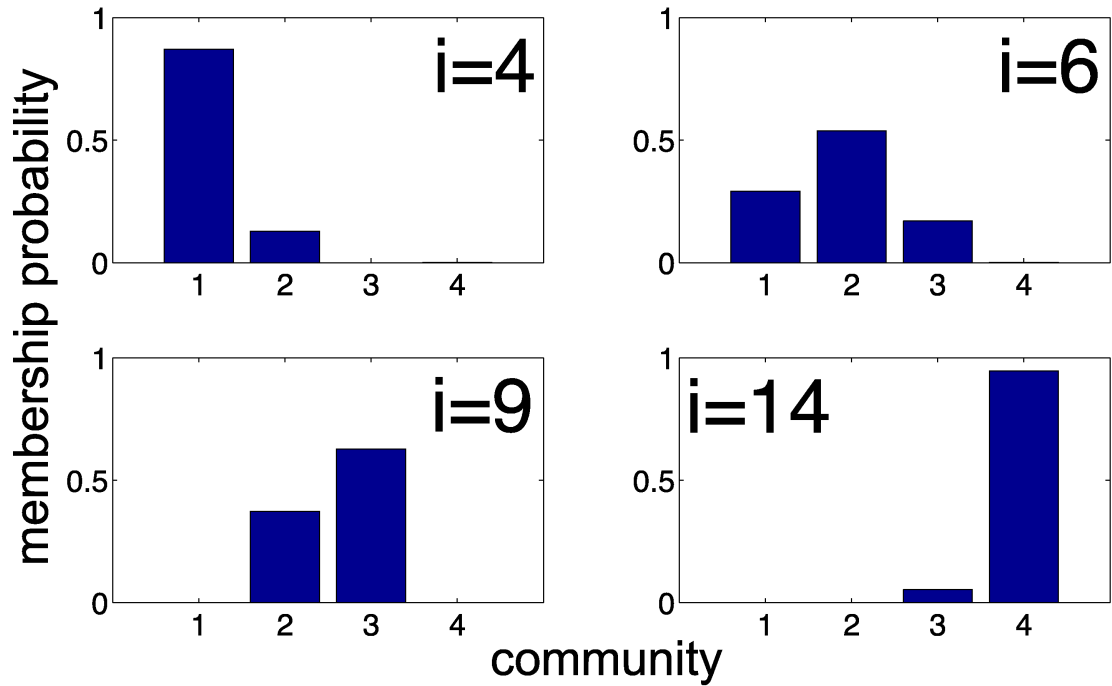


Figure 7: For each node of the toy graph our method gives a probability membership distribution; in the horizontal axis we enumerate each community appearing in Fig. 6 and each bar represents the likelihood that node i belongs to the respective module. For nodes that weakly communicate with other groups (such as 4 and 14) we see confident allocations, while individuals that lie on the boundary between communities (such as 6) have more entropic membership distribution.

tendency for cooperation, etc). Every individual contributes to a range of communities a certain amount of such energy, which can be also seen as his/her degree of membership. High-energy members can be regarded as *focal individuals* in a group, while social structures with members of uniform contribution can be regarded as *teams* that are held together because of equal participation of their members. Finally, under this framework we can identify highly social individuals, that belong to many groups with high amount of participation.

Having used a simple graph to illustrate the intuition behind NMF-based community detection, we proceed in the following section to demonstrate its performance on artificial problems of larger scale and complexity.

3.3 Benchmark datasets

A very popular evaluation methodology for a community detection algorithm is to test it against an artificial network with “observed” community structure and measure how well the algorithm extracts the underlying mesoscopic organisation. The partition quality is usually compared with the original using the popular Normalised Mutual Information (NMI) criterion [9].

We start with arguably the most popular type of benchmark problem: realisations of the Newman-Girvan random graph [14] (NG graph). We generate networks with $N = 128$ nodes and $C = 4$ communities with $n = 32$ nodes each where each one has an average degree of $\langle k \rangle = 16$. By manipulating the expected *inter-community* degree $\langle k_{\text{out}} \rangle$ of nodes we test our algorithm against various levels of community cohesiveness. As seen in Figures 8 and 9 NMF produces state-of-the-art performance in extracting the original modules for any degree of fuzziness in the artificial network, outperforming the popular Spectral Partitioning and Hierarchical Clustering (complete linkage - angular distance) method and having similar performance to Extremal Optimisation.

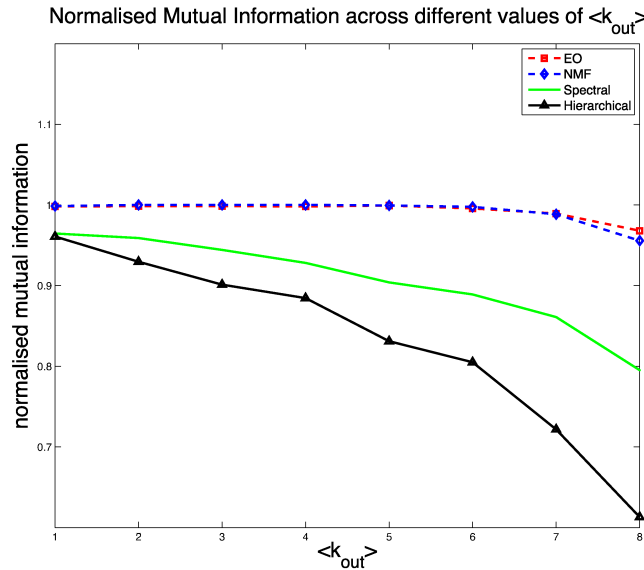


Figure 8: Normalised Mutual Information and modularity across different levels of community cohesion in a Newman-Girvan random network. We compare our method (NMF) against other popular community detection methodologies such as Extremal Optimization (EO), Spectral Partitioning (Spectral) and Hierarchical Clustering (Hierarchical).

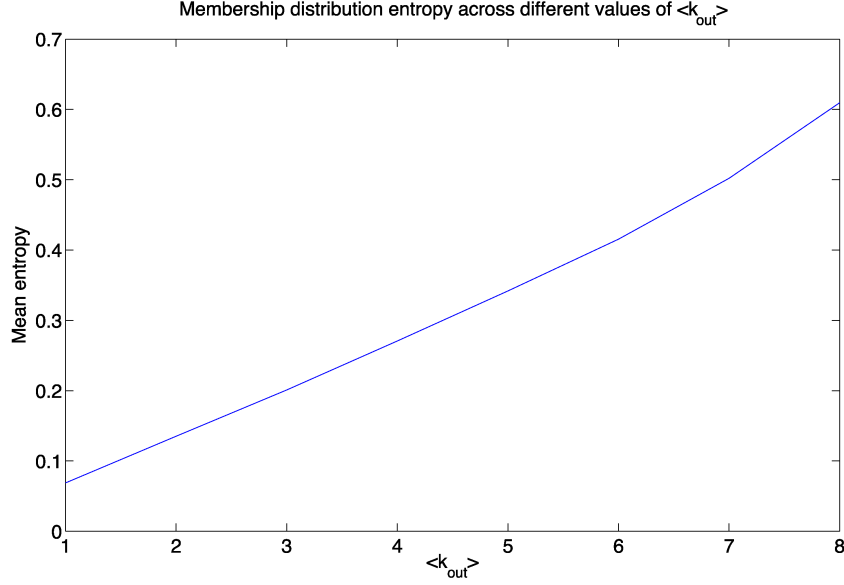


Figure 9: Mean entropy (in bits) of NMF node membership probabilities for decreasing levels of community cohesion in a Newman-Girvan random graph. We notice that NMF can describe the ever-increasing fuzziness of the NG graph in terms of decreasing node allocation confidence.

Although the NG graph is a very popular benchmark problem, it has been heavily criticised [20] for not reflecting the properties of real-world networks; NG graph realisations are small in size, with a fixed number of communities and fixed community populations while the degree distributions are uniform. For those reasons, Lancichinetti and Fortunato proposed a new class of benchmark problems [20] (which we shall refer to them as LF graphs) that produce networks of any size, with *power-law* degree and community size distributions. The community cohesiveness is controlled by a *mixing parameter* μ_t , that signifies the expected fraction of intercommunity links per node. For the case of weighted LF graphs, we have a similar parameter μ_w that controls the strength allocation of a node between same-community members and outsiders.

For the purposes of our experiments, we generated a variety of such networks with $N = 1000$ nodes and different parameters regarding the average degree $\langle k \rangle$ and the exponents γ_1, γ_2 of the degree and community size distributions. By starting with a small mixing parameter $\mu_t = 0.1$ and for each 0.1-step up to $\mu_t = 0.6$ (from ‘clear’ to ‘fuzzy’ community structure), we generate 100 realisations of the LF graph and monitor the module recognition performance of NMF using the popular normalised mutual information criterion. For the case of weighted LF graphs, we manipulate both mixing parameters at the same time, therefore $\mu_t = \mu_w$. The results, both for binary and weighted networks and for different configuration parameters are shown in Fig. 10 and 11.

3.4 Real-world datasets

In this section we present the performance of NMF on a variety of popular community detection problems. For these networks we have no “observed solution”, therefore we measure the performance of our algorithm using the very popular Newman-Girvan *modularity* Q [27]. In Table 1 we present a list of our datasets, along with their number of nodes N and edges M . Our algorithm is compared on the same data with Extremal Optimisation (EO) [10] and the Louvain [5] methods. We note other methods such as Spectral Partitioning and

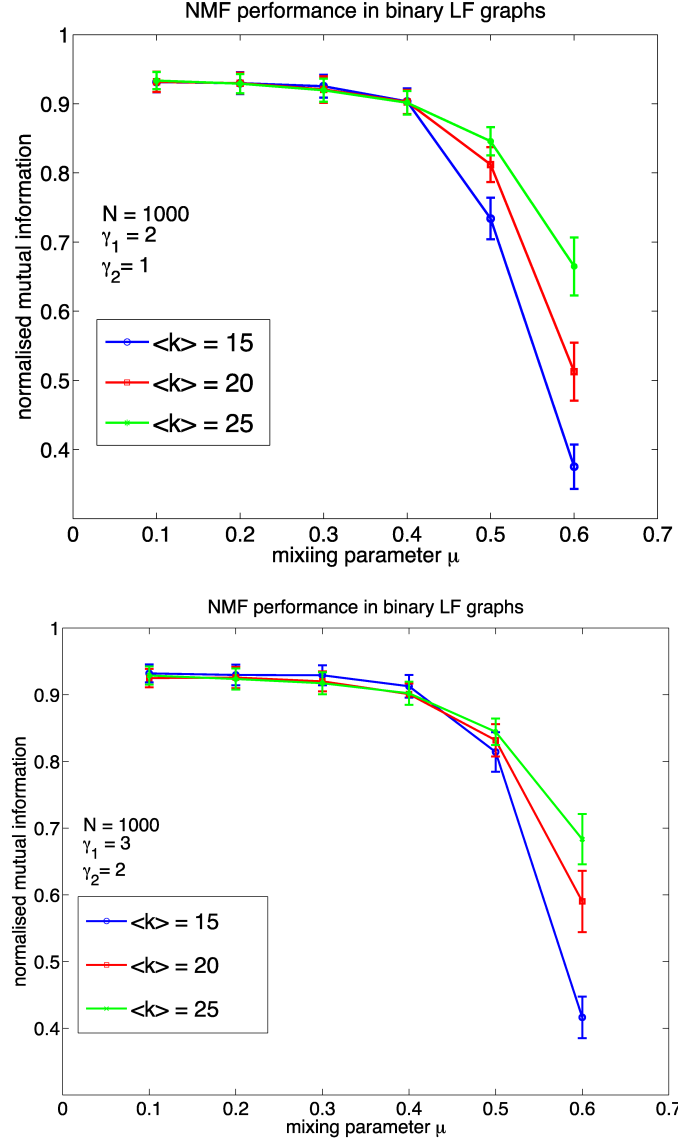


Figure 10: Normalised Mutual Information across different levels of community cohesion in binary LF random networks. We start with a very cohesive network (low mixing parameter μ) and proceed by making the network fuzzier. Each point in the graph represents the average over 100 realisations of an LF graph with the given parameters. The error bars represent one standard deviation.

Hierarchical Clustering algorithms give significantly worse performance than either NMF, EO or Louvain and these results are not presented here.

For each dataset we run NMF and EO 100 times with different random initialisations and monitored the values of *modularity* Q along with the number K^* of extracted communities. As previously detailed, for NMF initialisation, we assume a possible maximum number of communities, K , equal to the number of nodes $K = N$ (which is the maximum possible partition size for any network - though we find that running with

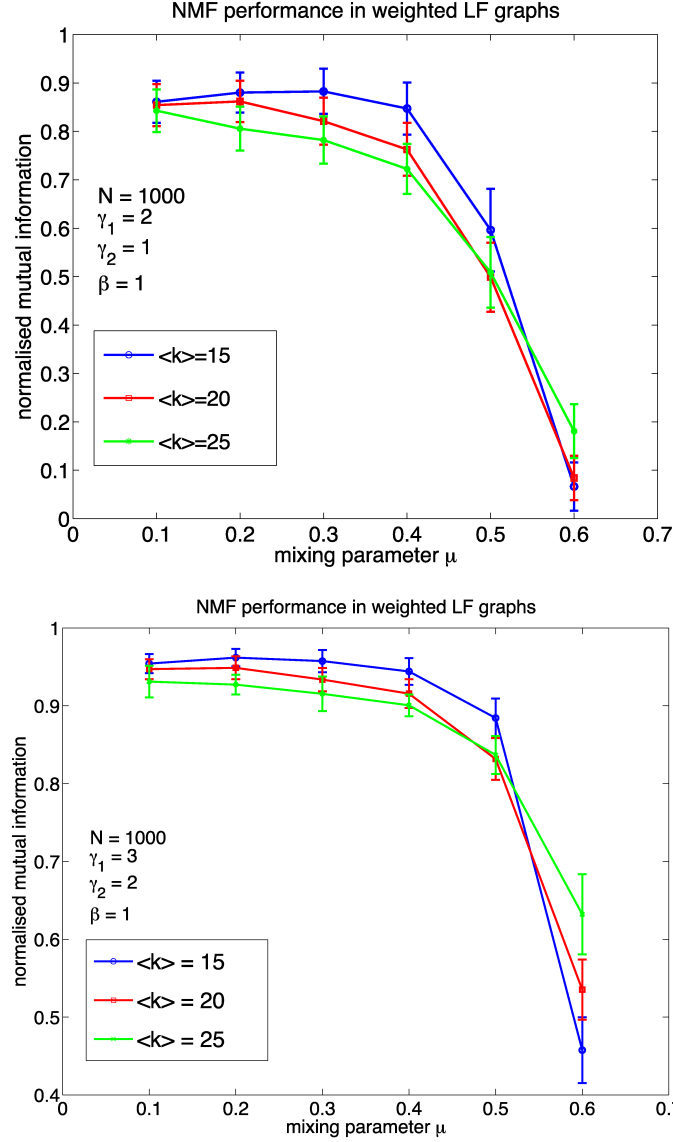


Figure 11: Normalised Mutual Information across different levels of community cohesion in weighted LF random networks. Again, we start with a very cohesive network and proceed by making the network fuzzier. In this case of weighted network, we have the same mixing parameter for both intercommunity degrees and strengths ($\mu_t = \mu_w$). Each point in the graph represents the average over 100 realisations of an LF graph with the given parameters and the error bars represent our standard deviation.

a lower value is preferred, as this reduces computation) and the ‘effective’ number of communities K^* is then inferred from the data. The Louvain method has a very stable behaviour across different runs so we have omitted the standard deviation of modularity and community sizes for each dataset. The algorithmic complexity of our approach is $\mathcal{O}(NK)$, as compared to $\mathcal{O}(N^2 \log N)$ for EO [9]. We also note that, in practice, as EO requires stochastic steps, the run times of the two algorithms differ even more significantly. In the majority of

Table 1: Real-world datasets

Dataset	N	M	weighted?
Dolphins [23]	62	159	no
Books US Politics [19]	105	441	no
Les Miserables [18]	77	254	yes
College Football [14]	115	613	no
Jazz Musicians [15]	198	2742	no
C. elegans metabolic [1]	453	2025	no
Network Science [27]	1589	2742	yes
Facebook Caltech [35]	769	16656	no

applications, the maximum likely number of communities $K \ll N$ and so our approach can be very efficient and competitive against the Louvain method.

Table 2: Modularity results against Extremal Optimisation and Louvain method

Dataset	NMF	EO	Louvain
Dolphins	0.47 ± 0.03	0.51 ± 0.01	0.52
Books US Politics	$0.52 \pm \epsilon$	0.48 ± 0.01	0.50
Les Miserables	0.53 ± 0.02	0.53 ± 0.01	0.57
College Football	$0.60 \pm \epsilon$	0.58 ± 0.01	0.60
Jazz Musicians	0.43 ± 0.01	0.42 ± 0.01	0.44
C. elegans metabolic	0.36 ± 0.01	0.40 ± 0.09	0.43
Network Science	0.83 ± 0.01	0.86 ± 0.01	0.95
Facebook Caltech	0.38 ± 0.01	0.37 ± 0.01	0.37

Table 3: NMF community sizes compared to Extremal Optimisation and Louvain method

Dataset	NMF	EO	Louvain
Dolphins	6.67 ± 0.83	4 ± 0	5
Books US Politics	6.23 ± 0.62	4.04 ± 0.4	3
Les Miserables	9.97 ± 0.78	4.96 ± 1.72	6
College Football	8.86 ± 0.79	8 ± 0	10
Jazz Musicians	8.57 ± 8.89	4 ± 0	4
C. elegans metabolic	15.69 ± 1.14	7.96 ± 1.06	10
Network Science	342.53 ± 5.28	58.24 ± 12.36	418
Facebook Caltech	24.28 ± 1.72	6.84 ± 1.82	10

In Table 2 we present our experimental results for each dataset of Table 1. We use the popular Newman-Girvan modularity Q as a performance measure of partition quality and we also present the number of identified communities. As Q can not account for the overlapping nature of communities, we use ‘greedy allocation’ i.e we assign a node to the module with the highest probability of membership. The results of NMF are presented alongside the very popular Extremal Optimisation and Louvain method for comparative analysis. From Table 2 we can see that our approach performs competitively yet is not an algorithm designed with the aim of maximising modularity, unlike either EO or the Louvain methods. Additionally, it has the advantage of providing

probabilistic outputs for community membership (therefore achieving soft partitioning) and having *low computational overhead*. Finally, NMF does not suffer from the resolution limit [13] of modularity optimisation methods such as EO, where smaller groups are merged together [30] [13], ending up with smaller number of communities, as seen in Table 3.

4 Conclusions

In this work we described a novel approach to community detection that adopts the Bayesian non-negative matrix factorisation model of [33] to achieve soft-partitioning of the network, assigning each node a probability of membership over all the extracted communities. That allows us not only to capture the fuzziness of the network (via the entropy of the membership distribution) but also to improve network cartography techniques [16] by identifying central and peripheral nodes in modules. Network visualization tools can also be improved in this manner. The approach is computationally efficient and offers performance comparable to state-of-the-art methods. Indeed the performance advantages for large data sets allow the NMF approach to be run many times compared to a single run of competing approaches. Clearly this allows for the selection of the best performing run, or a small ensemble of high-modularity solutions.

5 Future work

Future application work in this area addresses the analysis of a large zoological data set of interactions between members of a population of wild birds. As significant data exists and secondary verification data has been collated, such as breeding pair identifications etc. this offers a unique chance to verify any relationships that our approach detects.

Work is currently underway to allow this model to be extended to incorporate dynamics such that non-stationary, time-varying, community relationships may be tracked. We have not discussed the handling of missing data in this paper, but taking missing observations into account may be readily handled in our approach and this is detailed in [6]. As mentioned in the paper, more complex priors over the set of β_k would allow for domain knowledge to be incorporated and for correlated structures to be correctly dealt with.

6 Acknowledgements

The authors would like to thank Nick Jones, Mason Porter and Mark Ebden for valuable comments. Ioannis Psorakis is funded from a grant via Microsoft Research, for which we are most grateful.

References

- [1] http://deim.urv.cat/~aarenas/data/xarxes/celegans_metabolic.
- [2] J. M. Bernardo and A. F. M. Smith. *Bayesian Theory*. John Wiley, 1994.
- [3] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. In *Computational Statistics and Data Analysis*, pages 155–173, 2006.
- [4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, first edition, 2007.

- [5] V. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *J. Stat. Mech.*, 2008:P10008, October 2008.
- [6] A. T. Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Computational Intelligence and Neuroscience*, 2009(785152), 2009.
- [7] R. A. Choudrey and S. J. Roberts. Variational mixture of bayesian independent component analyzers. *Neural Computation*, 15(1):213–252, 2003.
- [8] P. Comon. Independent component analysis, a new concept? *Signal Processing.*, 36(3):287–314, 1994.
- [9] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas. Comparing community structure identification. *J. Stat. Mech.*, 2005(09):P09008, 2005.
- [10] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Phys. Rev. E*, 72(2):027104, 2005.
- [11] I. Farkas, D. Abel, G. Palla, and T. Vicsek. Weighted network modules. *New Journal of Physics*, 9(6):180, 2007.
- [12] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, February 2010.
- [13] S. Fortunato and M. Barthelemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.
- [14] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *PNAS*, 99(12):7821–7826, 2002.
- [15] P. Gleiser and L. Danon. Community structure in jazz. *Advances in Complex Systems*, 6(4):565–573, 2003.
- [16] R. Guimera and L. A. N. Amaral. Cartography of complex networks: modules and universal roles. *J. Stat. Mech.*, 2005(2):P02001, 2005.
- [17] I. T. Jolliffe. *Principal Component Analysis*. Springer Verlag, 1986.
- [18] D.E. Knuth. The stanford graphbase: A platform for combinatorial computing, 1993.
- [19] V. Krebs. <http://www.orgnet.com/>.
- [20] A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Phys. Rev. E*, 80(1):016118, 2009.
- [21] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorisation. *Nature*, 401:788–791, October 1999.
- [22] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *In NIPS*, pages 556–562. MIT Press, 2000.
- [23] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

- [24] M. E. J. Newman. The structure and function of complex networks. *SIAM Rev.*, 45(2):167–256, 2003.
- [25] M. E. J. Newman. Modularity and community structure in networks. *PNAS*, 103(23):8577–8582, 2006.
- [26] M. E. J. Newman. *Networks: an Introduction*. Oxford University Press, 2010.
- [27] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2):026113, 2004.
- [28] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature Letters*, 435(7043):814–818, 2005.
- [29] W. Penny and S. J. Roberts. Bayesian multivariate autoregressive models with structured priors. *IEE Proceedings on Vision, image and signal processing*, 149(1):33–41, 2002.
- [30] M. A. Porter, J. P. Onnela, and P. J. Mucha. Communities in networks. *Notices of the American Mathematical Society*, 56(9):1082–1097 and 1164–1166, 2009.
- [31] S. Roberts and R. Everson. *Independent Component Analysis: principles and practice*. Cambridge University Press, 2001.
- [32] S. J. Roberts and R. A. Choudrey. Bayesian independent component analysis with prior constraints: An application in biosignal analysis. In Joab Winkler, Mahesan Niranjan, and Neil Lawrence, editors, *Deterministic and Statistical Methods in Machine Learning*, volume 3635 of *Lecture Notes in Computer Science*, pages 159–179. Springer Berlin / Heidelberg, 2005.
- [33] V. Tan and C. Févotte. Automatic relevance determination in nonnegative matrix factorization. In *SPARS09 - Signal Processing with Adaptive Sparse Structured Representations (2009)*, pages 1–19, 2009.
- [34] M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- [35] A. L. Traud, E. D. Kelsic, P. J. Mucha, and M. A. Porter. Community structure in online collegiate social networks. arXiv:0809.0960, 2008.